

AN END-TO-END MACHINE LEARNING SYSTEM FOR HARMONIC ANALYSIS OF MUSIC

Yizhao Ni, Matt Mcvicar, and Tijl De Bie
Intelligent Systems Lab
Department of Engineering Mathematics
University of Bristol
U. K.

Raul Santos-Rodriguez
Signal Theory and Communications Department
Universidad Carlos III de Madrid
Spain

ABSTRACT

We present a new system for simultaneous estimation of keys, chords, and bass notes from music audio. It makes use of a novel chromagram representation of audio that takes perception of loudness into account. Furthermore, it is fully based on machine learning (instead of expert knowledge), such that it is potentially applicable to a wider range of genres as long as training data is available. As compared to other models, the proposed system is fast and memory efficient, while achieving state-of-the-art performance.

1. INTRODUCTION

Chords, along with the key and bassline, are essential mid-level features of western tonal music, and their evolution is fundamental to musical analysis. In recent years, audio chord transcription and tonal key recognition have been very active fields [2, 4, 9–11, 13, 15, 18], and the increasing popularity of Music Information Retrieval (MIR) with applications using mid-level tonal features has established chord and key recognitions as useful and challenging tasks (see also e.g. the MIREX competitions).

Since chords and keys are musical attributes closely related to each other in western tonal music [8], the idea to learn both progressions of a song simultaneously comes naturally. In general, such key/chord recognition systems are implemented using a HMM-like approach, based on a set of features extracted from the audio signal. A well-established audio feature for harmonic analysis is the *chromagram* [6]. It is a 12-dimensional representation of the harmonic content of the audio signal segmented into so-called *frames*, and it reflects the distribution of energy along pitch classes. In this paper the chromagram for the audio signal \mathbf{x} is denoted as $\bar{\mathbf{X}} \in \mathbb{R}^{12 \times T}$, with T indicating the number of frames.

An HMM [17] commonly regards chromagrams and annotations as *Observed* and *Hidden* variables respectively. Let $\mathbf{k} \in \mathcal{A}_k^{1 \times T}$ and $\mathbf{c} \in \mathcal{A}_c^{1 \times T}$ be the key and the chord

annotations of \mathbf{x} , where \mathcal{A}_k and \mathcal{A}_c represent the alphabets of keys and chords respectively. HMMs can then be used to formalize a probability distribution $P(\mathbf{k}, \mathbf{c}, \bar{\mathbf{X}} | \Theta)$ jointly for the chromagram feature vectors $\bar{\mathbf{X}}$ and the annotations, with Θ representing the parameters of this distribution. Given an HMM with optimal parameters Θ^* , the key/chord recognition task is equivalent to finding $\{\mathbf{k}^*, \mathbf{c}^*\}$ that maximize the joint probability $\{\mathbf{k}^*, \mathbf{c}^*\} = \arg \max_{\mathbf{k}, \mathbf{c}} P(\mathbf{k}, \mathbf{c}, \bar{\mathbf{X}} | \Theta^*)$.

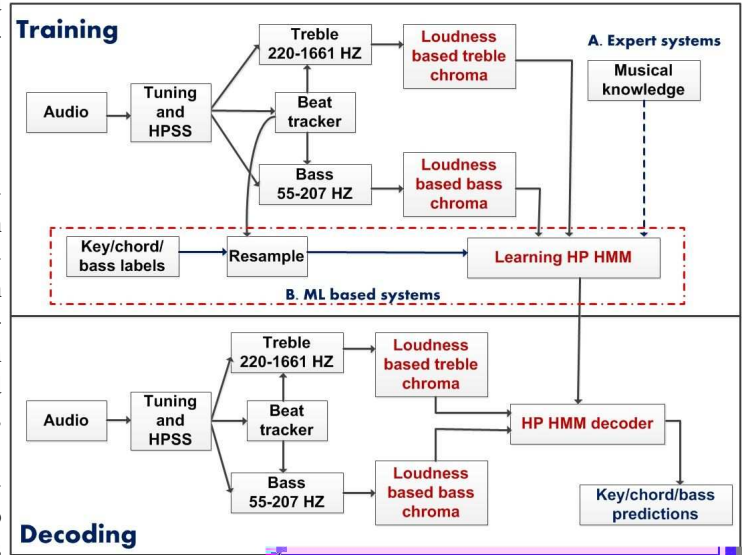


Figure 1. The learning procedure (via Approach B) of the proposed Harmony Progression (HP) system. The blocks in red show the novelties of the system.

Some existing key/chord recognition systems are based on Machine Learning (ML), where parameters are learned from a fully annotated training data set of features, keys and chords: $\{\mathcal{X}, \mathcal{K}, \mathcal{C}\} = \{\bar{\mathbf{X}}^n \in \mathbb{R}^{12 \times T_n}, \mathbf{k}^n \in \mathcal{A}_k^{1 \times T_n}, \mathbf{c}^n \in \mathcal{A}_c^{1 \times T_n}\}_{n=1}^N$ (Approach B in Figure 1) [9]. However, most approaches are based at least partially on expert knowledge, where parameters are set on the basis of music theoretic knowledge of the developers (Approach A in Figure 1) [2, 10, 11, 13, 15, 18]. For example, the key and chord transition parameters are set by hand, usually informed by perceptual key-to-key and chord-to-key relationships [8]. This contrasts with a clear tendency in Artificial Intelligence research to move away from systems based on expert knowledge to ML systems, e.g. in speech recognition,

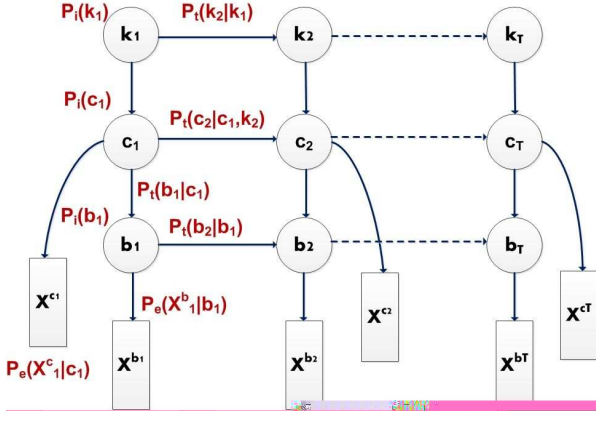


Figure 2. The HMM topology of the HP system. The probabilities in red are parameters of the system, which are learnt via maximum likelihood estimation (MLE).

machine translation, computer vision, etc. We start from the premise that the key/chord recognition task is not different and propose the *Harmony Progression (HP)* system for recognizing keys/chords from audio relying purely on ML techniques. The HP system is trained as illustrated in Figure 1 (Approach B) and the detailed HMM topology is depicted in Figure 2. Generally speaking, it is a simultaneous key/chord predictor that also identifies bass notes, going beyond most of the existing key/chord recognition systems [2, 9, 10, 13, 15, 18]. To our knowledge, the only system sharing a similar HMM topology is the expert knowledge based system proposed in [11] – the musical probabilistic model.

Compared with the MP system, the proposed HP system incorporates two additional major breakthroughs. Firstly, it utilizes a novel chromagram extraction method, supported with a well-founded physical interpretation. Secondly, our system is shown to be fast and memory-efficient in a case study. It also achieves an excellent tradeoff between performance and processing time in our experiments.

2. SYSTEM DESCRIPTION

2.1 Loudness based chromagram

Let $\mathbf{x} = [x_1, \dots, x_T]$ be an audio signal with x_t indicating the sample data of the t -th frame, then the chromagram extraction assigns attributes (e.g. power or amplitude) $\mathbf{X} \in \mathbb{R}^{S \times T}$ to a set of frequencies $F = \{f_1, \dots, f_S\}$ such that \mathbf{X} reflects the energy distribution of the audio along these frequencies. In order to capture musically relevant information, the frequencies are selected from the equal-tempered scale, which may be tuned [7] and vary between songs. Popular implementations of chromagram extraction are *fixed bandwidth Fourier* [6] and *constant Q* [1] transforms.

The above two chromagram systems represent the salience of pitch classes in terms of a power or amplitude spectrum. We note however that perception of loudness is not linearly proportional to the power or amplitude spectrum, and hence such chromagram representations do not accu-

ately represent human perception of the audio’s spectral content. Although there is an alternative chromagram that claimed to model human auditory sensitivity [16], the proposed framework is very primitive. The chromagram still uses spectrum as pitch energy and it just utilizes an arc-tangent function to mimic pitch perception without any rigorous reference. In fact, the empirical study in [5] showed that loudness is approximately linearly proportional to so-called *sound power level*, defined as \log_{10} of power spectrum. Therefore, we developed a novel *loudness based chromagram*, which uses the \log_{10} scale of power spectrum. Mathematically, a sound power level (SPL) matrix is of the form

$$\mathcal{L}_{s,t} = 10 \log_{10} \left(\frac{\|X_{s,t}\|^2}{p_{ref}} \right), \quad s = 1, \dots, S, t = 1, \dots, T,$$

where p_{ref} indicates the fundamental reference power and

$$X_{s,t} = \sum_{n=t-\frac{L_s}{2}}^{t+\frac{L_s}{2}} x_n w_n \exp \left(\frac{-2\pi s t}{L_s} \right)$$

is a constant Q transform with a frequency dependent bandwidth $L_s = Q \frac{SR}{f_s}^1$ and the hamming window w_n [1].

Furthermore, low/high frequencies require higher sound power levels for the same perceived loudness as mid-frequencies [5]. To compensate for this, we propose to use *A-weighting* [20] to transform the SPL matrix into a representation of the perceived loudness of each of the pitches:

$$\mathcal{L}'_{s,t} = \mathcal{L}_{s,t} + A(f_s), \quad s = 1, \dots, S, t = 1, \dots, T,$$

where

$$R_A(f_s) = \frac{12200^2 \cdot f_s^4}{(f_s^2 + 20.6^2) \cdot \sqrt{(f_s^2 + 107.7^2)(f_s^2 + 737.9^2)} \cdot (f_s^2 + 12200^2)}$$

$$A(f_s) = 2.0 + 20 \log_{10}(R_A(f_s)).$$

It is known that loudnesses are additive if they are not close in frequency [19]. This allows us to sum up loudness of sounds on the same pitch class, yielding:

$$X'_{p,t} = \sum_{s=1}^S \delta(M(f_s), p) \mathcal{L}'_{s,t}, \quad p = 1, \dots, 12, t = 1, \dots, T.$$

Here δ denotes an indicator function and

$$M(f_s) = \left(\left\lfloor 12 \log_2 \left(\frac{f_s}{f_A} \right) + 0.5 \right\rfloor + 69 \right) \bmod 12$$

with f_A denoting the reference frequency of the pitch A4 (440Hz in standard pitch). Finally, our loudness-based chromagram, denoted $\bar{X}_{p,t}$, is obtained by normalizing $X'_{p,t}$ using:

$$\bar{X}_{p,t} = \frac{X'_{p,t} - \min_{p'} X'_{p',t}}{\max_{p'} X'_{p',t} - \min_{p'} X'_{p',t}}.$$

Note that this normalization is invariant to the reference power and hence a specific p_{ref} is not required.

¹ Q is a constant resolution fact which can be tuned by the cross-validation technique and SR is the sampling rate of the audio signal.

2.2 HP HMM topology

The HP HMM topology consists of three hidden and two observed variables. The hidden variables correspond to the key \mathcal{K} , the chord \mathcal{C} and the bass annotations $\mathcal{B} = \{\mathbf{b}^n \in \mathcal{A}_b^{1 \times T_n}\}_{n=1}^N$. Under this representation, a chord is decomposed into two aspects: chord label and bass note. Take the chord A:maj/3 for example, the chord state is $c = \text{A:maj}$ and the bass state is $b = \text{C}\#$. Accordingly, the observed chromagrams are decomposed into two parts: the treble chromagram $\bar{\mathbf{X}}^c$ which is emitted by the chord sequence \mathbf{c} and the bass chromagram $\bar{\mathbf{X}}^b$ which is emitted by the bass sequence \mathbf{b} . The reason of applying this decomposition is that different chords can have the same bass note, resulting in similar chromagrams in low frequency domain.

Under this framework, the set Θ of a HP HMM has the following parameters

$$\Theta = \{p_i(k_1), p_i(c_1), p_i(b_1), p_t(k_t|k_{t-1}), p_t(c_t|c_{t-1}, k_t), p_t(b_t|c_t), p_t(b_t|b_{t-1}), p_e(\bar{\mathbf{X}}_t^c|c_t), p_e(\bar{\mathbf{X}}_t^b|b_t)\},$$

where p_i , p_t and p_e denote the initial, transition and emission probabilities respectively. The joint probability of the feature vectors $\{\bar{\mathbf{X}}^c, \bar{\mathbf{X}}^b\}$ and the corresponding annotation sequences $\{\mathbf{k}, \mathbf{c}, \mathbf{b}\}$ of a song is then given by the formula²

$$P(\bar{\mathbf{X}}^c, \bar{\mathbf{X}}^b, \mathbf{k}, \mathbf{c}, \mathbf{b}|\Theta) = p_i(k_1)p_i(c_1)p_i(b_1) \prod_{t=2}^T p_t(k_t|k_{t-1}) p_t(c_t|c_{t-1}, k_t) p_t(b_t|c_t) p_t(b_t|b_{t-1}) p_e(\bar{\mathbf{X}}_t^c|c_t) p_e(\bar{\mathbf{X}}_t^b|b_t).$$

The initial probabilities $p_i(\star)$ can be learnt via *maximum likelihood estimation* (MLE). For example, $p_i(c) = \frac{\#(c_1=c)}{\#c_1} \forall c \in \mathcal{A}_c$, where $\#$ indicates the number of.

For the transitions, $p_t(c|\bar{c}, k)$ represents the probability of a chord change under a certain key. Since the chord transition is strongly influenced by the underlying key [13], this probability is modelled as key dependent. Under the assumption that relative chord transitions are key independent, we transposed all sequences to a common key k and learn $p_t(c|\bar{c}, k)$ from the transposed sequences. This allowed us to get 12 times as much information from the data source and the MLE solution is

$$p_t(c|\bar{c}, k) = \frac{\#(c_t = c \& c_{t-1} = \bar{c} \& k_t = k)}{\sum_{c'} \#(c_t = c' \& c_{t-1} = \bar{c} \& k_t = k)}, \forall c, \bar{c}, k.$$

Similarly, $p_t(k|\bar{k})$ is applied to model key changes during a song. $p_t(b|c)$ models the probability of a bass note under a chord label so as to capture chord inversions. A transition link $p_t(b|\bar{b})$ is also added, with the purpose of modelling the continuity of bass notes and capturing ascending and descending bassline progressions. These parameters are learnt via MLE, e.g. $p_t(k|\bar{k}) = \frac{\#(k_t=k \& k_{t-1}=\bar{k})}{\sum_{k'} \#(k_t=k' \& k_{t-1}=\bar{k})}, \forall k, \bar{k} \in \mathcal{A}_k$.

Finally, emission probabilities $p_e(\bar{\mathbf{X}}_t^c|c_t)$ and $p_e(\bar{\mathbf{X}}_t^b|b_t)$ are modelled as 12-dimensional Gaussians, of which the mean vectors and covariance matrices are learnt via MLE as well.

² Note that we use $p_t(b_t|b_{t-1}, c_t) = p_t(b_t|c_t)p_t(b_t|b_{t-1})$, which from a purely probabilistic perspective is not correct. However, this simplification reduces computational and statistical cost and results in better performance in practice.

2.3 Search space reduction

Given the optimal parameters Θ^* via MLE, the decoding task can be formalized as the computation of the key, chord and bass sequences $\{\mathbf{k}^*, \mathbf{c}^*, \mathbf{b}^*\}$ that maximize the joint probability $\{\mathbf{k}^*, \mathbf{c}^*, \mathbf{b}^*\} = \arg \max_{\mathbf{k}, \mathbf{c}, \mathbf{b}} P(\bar{\mathbf{X}}^c, \bar{\mathbf{X}}^b, \mathbf{k}, \mathbf{c}, \mathbf{b}|\Theta^*)$.

This task can be solved using the *Viterbi* algorithm [17], whose computational complexity is $O(|\mathcal{A}_k|^2|\mathcal{A}_c|^2|\mathcal{A}_b|^2|T|)$. This is a huge search space, especially when one would like to use a large chord vocabulary [11]. In order to reduce the decoding time, we propose three constraints on the search space:

2.3.1 Key transition constraint

Music theory dictates that not all key changes are equally likely. If a song does change key, the modulation is most likely to move to a related key [8]. Thus, we suggest to rule out a priori the key transition that are seen the least often in the training set. Formally, this can be done by constraining the key transition probability as

$$p'_t(k|\bar{k}) = \begin{cases} p_t(k|\bar{k}) & \text{if } \#(k_t = k \& k_{t-1} = \bar{k}) > \gamma \\ 0 & \text{otherwise} \end{cases},$$

where γ is a positive integer indicating the threshold.

2.3.2 Chord to bass transition constraint

Similar to the key transition constraint, we can also constrain the chord to bass transitions. A constraint is imposed on $p_t(b|c)$ such that the bass notes can only be one of τ ($\tau \leq 12$) candidates for a given chord. The frequencies of each chord-to-bass emission are ranked and only the most common τ are permissible. Mathematically:

$$p'_t(b|c) = \begin{cases} p_t(b|c) & \text{if } b \text{ is one of the top } \tau \text{ bass notes for } c \\ 0 & \text{otherwise} \end{cases}.$$

When $\tau = 3$, the constraint is equivalent to using root position, first and second inversions of a chord.

2.3.3 Chord alphabet constraint (CAC)

It is unlikely that all chords will be used in a single song. Therefore, if it is possible to find out which chords are used in a song, we will be able to constrain the chord alphabet without loss of performance. One heuristic method is to utilize two-stage predictions. In particular, using a simple HMM with only chords as the hidden chain, we first apply a max-Gamma decoder [17] to a song and obtain the most probable chords \mathcal{A}'_c . Then, we force the HP HMM chord transition probability to be zero for chords that are absent in this output:

$$p'_t(c|\bar{c}, k) = \begin{cases} p_t(c|\bar{c}, k) & \text{if } c, \bar{c} \in \mathcal{A}'_c \\ 0 & \text{otherwise} \end{cases}.$$

3. EXPERIMENTS

3.1 Audio dataset and ground truth annotations

The audio dataset used is the one used in the MIREX Chord Detection task 2010³, which contains 217 songs. The

³ http://www.music-ir.org/mirex/wiki/2010:Audio_Chord_Estimation

The information is quoted from [] (page 78).

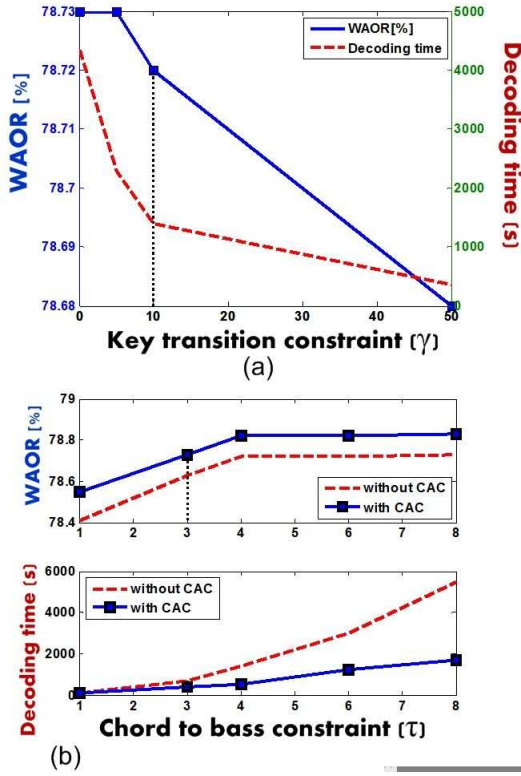


Figure 3. The performances and decoding times of HP using different search space reductions. The experiments in (a) were done without chord alphabet constraint and τ is fixed at 4. In (b), ‘CAC’ refers to chord alphabet constraint and the experiments were carried out with γ fixed at 10.

of-the-art MP model (Table 2). Encouragingly, HP consumes less memory and is faster, even using a slower CPU.

	Processing time (s)		Peak memory (G)	
	HP	MP	HP	MP
Song 1	58	131	0.48	6
Song 2	171	345	1.20	15

Table 2. The comparison of processing time and memory consumption between the HP and MP systems. Song 1 is “Ticket to Ride” (190s) and Song 2 is “I Want You (She’s So Heavy)” (467s). The MP results were performed on a computer running CentOS 5.3 with 8 Xeon X5577 cores at 2.93GHz, 24G RAM. HP was run on a CentOS 5.6 computer with Intel (R) X5650 cores at 2.67GHz, 24G RAM.

Since MP is not publicly available, we instead compared HP to Chordino [12] (denoted by CH) which uses the same NNLS chroma features as MP but a simpler model. Comparing with CH also seems more appropriate because its computation/memory cost is more reasonable and in line with HP. For HP, the parameters τ and γ are fixed at 3 and 10. All other parameters are trained using the whole dataset (denoted by HP-P). To assess generalization ability, we also computed the leave-one-out error for HP (denoted by HP-L). We used 3 performance metrics: chord

precision (CP), which scores 1 if the ground truth and predicted chords are identical and 0 otherwise (e.g. the score between A:maj/3 and A:maj is 0); note-based chord precision (NCP), which scores 1 if all notes are identical between ground truth and predicted chords and 0 otherwise (e.g. the score between A:maj/3 and A:maj is 1 but that between A:maj and A:maj7 is 0), and the MIREX ‘WAOR’ evaluation. All evaluations are performed with 1ms sampling rate, as used in MIREX 2010 competition. Tests were done on a MAC with an Intel Duo Core 2.4G CPU and 4G RAM.

Table 3 shows a very large improvement over the baseline CH, even on the MIREX-style evaluation. Moreover, the full chord HP-P system achieves a further improvement on WAOR over the HP-P in the major/minor chord prediction task, again indicating that increasing the complexity of models helps harmonic estimation. Meanwhile, we found the cause of the low performance of CH is that it predicted many complex chords (notably 7ths). This is a good strategy for the MIREX evaluation, that only measures the overlap recall between notes in predicted and ground truth chords. However, it does adversely affect the performances measured using CP and NCP. Comparing the processing time, our system is slightly slower due to the separate calculation of bass and treble chromagrams. However, the decoding process is very fast and thus the system is still easy to apply to real world harmonic analysis tasks.

System	CP [%]	NCP [%]	WAOR [%]
CH	50.31	52.35	76.94
HP-L	63.63	65.24	81.05
HP-P	70.26	71.96	82.98

System	Processing time (s)	
	Feature extraction	Decoding
CH	9511	
HP	12756	818

Table 3. Performance (top) and processing time (bottom) for the baseline and HP systems on the full chord prediction task. Bold numbers refer to the best results. Note that for the CH system only the whole processing time is available.

4. CONCLUSIONS AND FUTURE WORK

In this paper we propose a novel key, chord and bass simultaneous recognition system – the HP system – that purely relies on ML techniques. The experimental results verify that the HP system can achieve the state-of-the-art performance on chord recognition, and it can be sped up significantly using the search space reduction techniques without severely decreasing the performance.

HP uses a novel chromagram extraction method, which is inspired by loudness perception studies and achieves better recognition performance. Secondly, HP purely relies on ML techniques, which provides more flexibility in its applications and promises further improvements if more

data becomes available. Finally, HP achieves an excellent tradeoff between performance and processing time, making it applicable to real world harmonic analysis tasks.

For future work, we aim to improve the processing time for chromagram extraction. This can be done by moving to faster programming languages such as C and C++. We will also move towards discriminative approaches using the same HMM topology, which might lead to a more robust and powerful harmonic analysis tool.

5. REFERENCES

- [1] J. Brown. Calculation of a constant q spectral transform. *Journal of the Acoustical Society of America*, 89(1):425–434, 1991.
- [2] B. Catteau, J. Martens, and M. Leman. A probabilistic framework for audio-based tonal key and chord recognition. In *Proc. of GfKI*, pages 637–644, 2006.
- [3] D. Ellis and G. Poliner. Identifying ‘cover songs’ with chroma features and dynamic programming beat tracking. In *Proc. of ICASSP*, pages 1429–1433, 2007.
- [4] D. Ellis and A. Weller. The 2010 LABROSA chord recognition system. In *Proc. of ISMIR (MIREX)*, 2010.
- [5] H. Fletcher. Loudness, its definition, measurement and calculation. *Journal of the Acoustical Society of America*, 5(2):82, 1933.
- [6] T. Fujishima. Real time chord recognition of musical sound: a system using common lisp music. In *Proc. of ICMC*, pages 464–467, 1999.
- [7] C. Harte and M. Sandler. Automatic chord identification using a quantised chromagram. In *Proc. of the Audio Engineering Society*, 2005.
- [8] C. L. Krumhansl. *Cognitive foundations of musical pitch*. Oxford University Press, 1990.
- [9] K. Lee and M. Slaney. A unified system for chord transcription and key extraction using hidden markov models. In *Proc. of ISMIR*, 2007.
- [10] K. Lee and M. Slaney. Acoustic chord transcription and key extraction from audio using key-dependent hmms trained on synthesized audio. *The IEEE Transactions on Audio, Speech and Language Processing*, 2008.
- [11] M. Mauch. *Automatic chord transcription from audio using computational models of musical context*. PhD thesis, Queen Mary University of London, 2010.
- [12] M. Mauch and S. Dixon. Approximate note transcription for the improved identification of difficult chords. In *Proc. of ISMIR*, 2010.
- [13] K. Noland and M. Sandler. Key estimation using a hidden markov model. In *Proc. of ISMIR*, 2006.
- [14] N. Ono, K. Miyamoto, J. Roux, H. Kameeoka, and S. Sagayama. Separation of a monaural audio signal into harmonic/percussive components by complimentary diffusion on spectrogram. In *Proc. of EUSIPCO*, 2008.
- [15] H. Papadopoulos and G. Peeters. Local key estimation based on harmonic and metric structures. In *Proc. of DAFX*, 2009.
- [16] S. Pauws. Musical key extraction from audio. In *ISMIR*, 2004.
- [17] L. R. Rabiner. A tutorial on hidden markov models and selected application in speech recognition. In *Proc. of the IEEE*, 1989.
- [18] T. Rocher, M. Robine, P. Hanna, L. Oudre, Y. Grenier, and C. Févotte. Concurrent estimation of chords and keys from audio. In *Proc. of ISMIR*, 2010.
- [19] T. D. Rossing. *The science of sound (second edition)*. Addison-Wesley, 1990.
- [20] M. T. Smith. *Audio engineer’s reference book*. Focal Press, 1999.